Neural Computing and Applications
https://doi.org/10.1007/s00521-021-06164-7

ORIGINAL ARTICLE

Automatic chronic degenerative diseases identification using enteric nervous system images

Gustavo Z. Felipe[1] · Jacqueline N. Zanoni[1] · Camila C. Sehaber-Sierakowski[1] · Gleison D. P. Bossolani[1] · Sara R. G. Souza[1] · Franklin C. Flores[1] · Luiz E. S. Oliveira[2] · Rodolfo M. Pereira[3] · Yandre M. G. Costa[1]

Received: 11 November 2020 / Accepted: 24 May 2021

Abstract
Studies recently accomplished on the Enteric Nervous System have shown that chronic degenerative diseases affect the Enteric Glial Cells (EGC) and, thus, the development of recognition methods able to identify whether or not the EGC are affected by these type of diseases may be helpful in its diagnoses. In this work, we propose the use of pattern recognition and machine learning techniques to evaluate if a given animal EGC image was obtained from a healthy individual or one affect by a chronic degenerative disease. In the proposed approach, we have performed the classification task with handcrafted features and deep learning-based techniques, also known as non-handcrafted features. The handcrafted features were obtained from the textural content of the ECG images using texture descriptors, such as the Local Binary Pattern (LBP). Moreover, the representation learning techniques employed in the approach are based on different Convolutional Neural Network (CNN) architectures, such as AlexNet and VGG16, with and without transfer learning. The complementarity between the handcrafted and non-handcrafted features was also evaluated with late fusion techniques. The datasets of EGC images used in the experiments, which are also contributions of this paper, are composed of three different chronic degenerative diseases: Cancer, Diabetes Mellitus, and Rheumatoid Arthritis. The experimental results, supported by statistical analysis, show that the proposed approach can distinguish healthy cells from the sick ones with a recognition rate of 89.30% (Rheumatoid Arthritis), 98.45% (Cancer), and 95.13% (Diabetes Mellitus), being achieved by combining classifiers obtained on both feature scenarios.

Keywords Degenerative chronic diseases · Enteric glial cells · Pattern recognition · Deep learning · Machine learning

1 Introduction

The Enteric Nervous System (ENS), which the small intestine is dependent of, controls the digestive tract, coordinating different movement patterns such as: fast propulsion of content (peristalsis), mixing movements (segmentation), slow propulsion and retropulsion (expulsion of harmful substances associated with vomiting) [16].

Two networks, or neural plexus, compose the main components of the ENS: (1) an plexus located between the longitudinal and circular muscle layers, called myenteric (or Auerbach's) plexus; and (2) the submucosal (or Meissner's) plexus, located in the submucosa. The myenteric plexus controls the gastrointestinal movements. While the submucosal plexus, overall, controls the gastrointestinal secretion and the local blood flow [16]. The Enteric Glial Cells (EGC) are another type of cells that can be found in the ENS as well. These ones play a vital role in the homeostasis of the gastrointestinal tract (GIT) functions [44].

Formerly, it was thought that the EGC worked only as a structural support to the neurons. But decisive studies carried out more recently [18, 44] verified that these cells

We thank the Brazilian Research Support Agency CNPq - National Council for Scientific and Technological Development.

✉ Gustavo Z. Felipe
pg402913@uem.br

[1] Universidade Estadual de Maringá (UEM), Av. Colombo 5790, 87020-900 Maringá, PR, Brazil

[2] Universidade Federal do Paraná (UFPR), Rua Cel. Francisco H. dos Santos 100, 81531-990 Curitiba, PR, Brazil

[3] Instituto Federal do Paraná (IFPR), R. Humberto de A. C. Branco 1575, 83330-200 Pinhais, PR, Brazil

Published online: 17 June 2021

also have other functions, contributing significantly to the neuronal maintenance, survival and function. In neurodegenerative processes, these cells play a role in the neuronal reconstruction, increasing their expression [41].

The immunostaining, i.e., an antibody-based method to detect a specific protein in a sample, of the S100 protein is used to identify the EGC. The S100 is a calcium binding protein located in the cytoplasm and/or nucleus of nerve and non-nerve tissues and can be expressed exclusively in the EGC. This one regulates the cytoskeleton's structure and function, as well as the calcium homeostasis in EGC's cytoplasm. It also presents neurotrophic properties that play a neuroprotective function [17].

The study of ENS cells is usually approached in preclinical research, aiming to experiment new methodologies and techniques in animals to be, later on, employed in humans. Resulting in the non-exposure of patients to the risk of death or permanent disability. Several projects involving the ENS have been developed by researchers in the field of neurogastroenterology [2, 34, 35]. In these works, the enteric neurons and EGC are studied in order to understand the impact suffered by these cells on different diseases, as well as analyze the performance of different treatments for them.

Usually, the enteric neurons and the EGC are preferable in such studies, because these kind of cells are all heavily affected by a considerable portion of chronic degenerative diseases. Thus, different studies may be taken on different diseases by evaluating a single kind of image sample.

Considering that the disease affects the EGC in shape and quantity, to ascertain the healthiness of a target animal, the researcher performed morphometric and quantitative analyses. These can be declared as exhaustive and time-consuming, once the lack of automation in the overall process, makes it extraordinarily manual and repetitive. It is reaffirming the relevance of developing computational models that execute such tasks automatically and with efficiency.

With that in mind, this work aimed to develop an approach for the automatic identification of chronic degenerative diseases in EGC animal images. Being capable of categorizing if an image sample from the ENS, evidencing the EGC, was obtained by a healthy or a sick animal.

The proposed method aims to classify the images based on the extraction of handcrafted and non-handcrafted (automated learned) features. The handcrafted features are the ones here extracted by texture descriptors, while the non-handcrafted features are obtained with the use of Convolutional Neural Networks (CNNs). We have also investigated their complementarity by combining the resulting classifiers from both scenarios through classifiers' combination techniques. As far as we know, this is the first work to deal with identifying chronic degenerative diseases on ENS images.

To experimentally evaluate the proposed approach, we have created three datasets with EGC images of rats affected by different chronic degenerative diseases: Diabetes Mellitus, Cancer (Walkers Tumor-256), and Rheumatoid Arthritis. Each dataset is composed of image samples, collected by the evidence of EGC from the myenteric plexus from control (healthy) animals and from animals that presented the target disease. The datasets are freely available for download and can also be considered as a contribution to this work.

By achieving this goal, we look forward to giving the ENS researchers a texture-based automatic alternative to perform the EGC image analysis and foment new computer science research with the ENS, considering its great unexplored potential. It is worth mentioning that this one may be expanded to automatically detect diseases in human histopathological and radiographic images, aiming to reduce the possible subjectivity in their analysis.

The remaining of this work is organized as follows: Sect. 2 presents the approach proposed in this work, describing the data augmentation protocol, feature extraction, classification and combination phases. Section 3 presents the three degenerative diseases datasets proposed in this work. Section 4 presents the experimental analysis of the proposed approach, which is subdivided in exploratory investigation, parameters, configurations and results. In Sect. 5, we present the analysis and discussion regarding the obtained results and, finally, in Sect. 6 we describe the concluding remarks and future works.

2 Proposed approach

By analyzing the content of the images investigated in this work, we can observe that texture is one of the primary visual content to be explored. In this way, we decided to organize our proposed approach mainly based on different strategies aimed at describing the textural content. Besides that, methods that can also cooperate to each other in the perspective of the combination of classifiers or representations. In this vein, we create descriptors founded both on the so-called handcrafted and non-handcrafted scenarios. An overview of the proposed approach can be seen in Fig. 1.

The handcrafted features correspond to features manually extracted, aiming to find the best representation of the addressed data through a process also known as feature engineering. These kinds of features were extracted in this work considering some widely and successfully used texture operators available in the literature. In total, three texture descriptors were broached: Local Binary Pattern

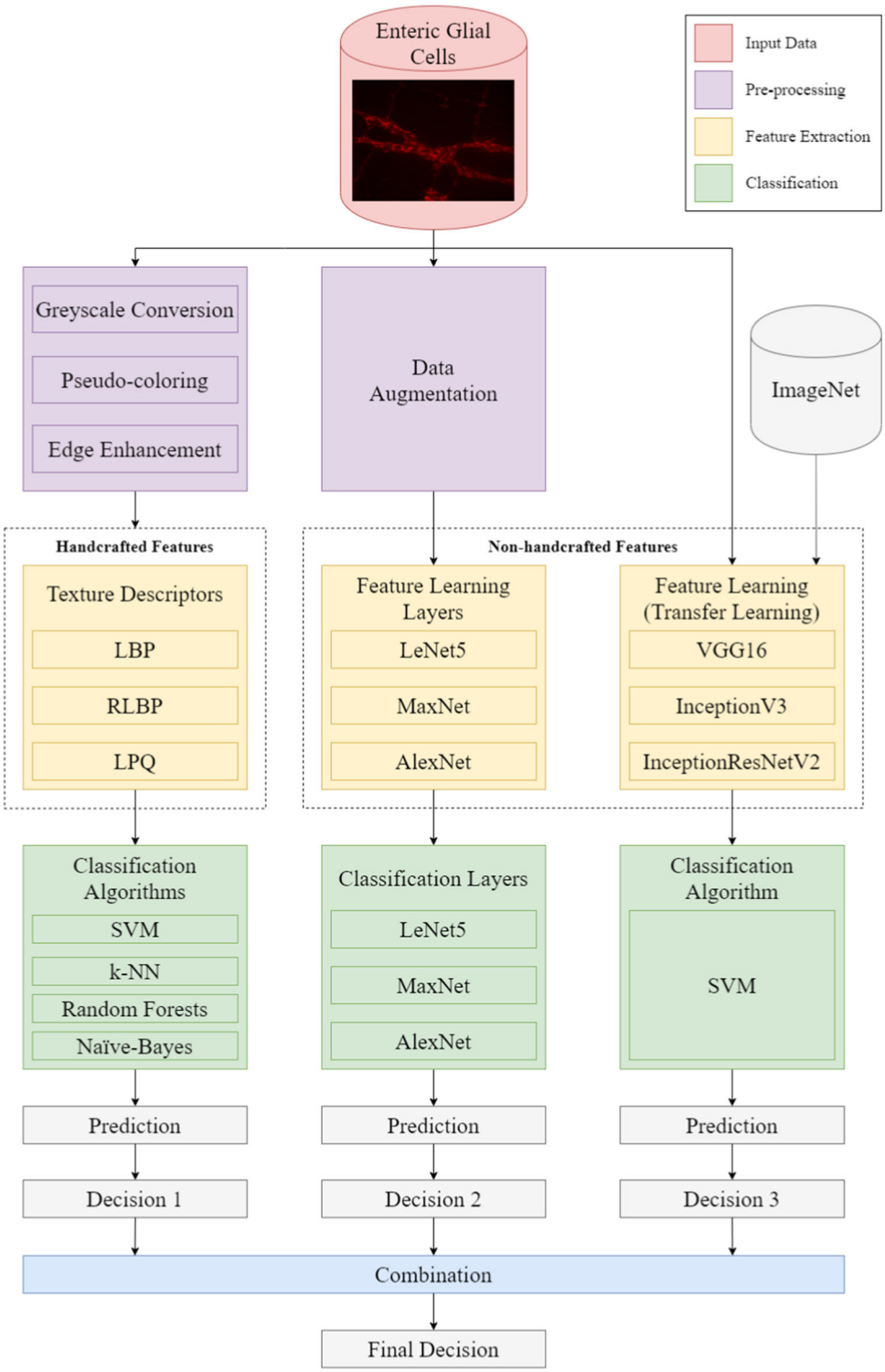

Fig. 1 Representation of the approach used in this work, in which handcrafted and non-handcrafted features were approached

(LBP) [31], Robust Local Binary Pattern (RLBP) [52] and Local Phase Quantization (LPQ) [32].

The handcrafted features are then used as input to well-known classification algorithms such as Support Vector Machines (SVM), Gradient Boosting (GB), Random Forests (RF), k-Nearest Neighbors (k-NN) and Naive-Bayes (NB) classification algorithms.

As aforementioned, the second scenario studied here extracts non-handcrafted features, i.e., features automatically extracted from the images, through feature learning techniques. In this work, we used three well-established

Convolutional Neural Network (CNN) architectures to obtain this kind of feature: LeNet5 [23], AlexNet [22] and MaxNet [40].

The concept of Transfer Learning was also experimented using pre-trained CNNs to extract features from the image samples. Thus, the CNN architectures VGG16 [45], InceptionV3 [49] and InceptionResNetV2 [48] were used. The feature learning layers of such models had their weights trained in the ImageNet dataset [10]. It is worth mentioning that the Chi-square test (X^2) was employed as a feature selection method, aiming to reduce the number of features extracted from these CNN architectures. Differing from the traditional use of transfer learning, instead of redesigning the CNN model's classification layers, in this work, we classified the resulting features using the SVM algorithm.

The estimation of probabilities generated from the classification experiments performed was then used to combine the resulting classifiers. Thus, the sum, product, and max classification rules, proposed by Kittler et al. [21], were applied aiming to take advantage of a possible complementarity between classifiers generated by the use of different features and techniques.

It is worth mentioning that the experiments used the Stratified K-Fold Cross-Validation technique to divide the dataset to keep the existing proportions of the problem's classes. In this work, the k value used was set to ten. More details about the concepts introduced here may be found in the following subsections.

2.1 Handcrafted modeling

This section describes the handcrafted modeling, which follows the Pattern Recognition framework to classify features extracted through feature engineering. For simplicity, we refer them as handcrafted features.

2.1.1 Data augmentation

In this work, some pre-processing techniques were explored, considering two main goals. The first one performs variations in the image samples' coloration. To achieve it, firstly, the image samples had their colors omitted by converting them to grayscale. Motivated by the fact that all image samples have a red tonality, implying that the two other channels of the RBG color system (blue and green) may not have any influence when extracting the handcrafted features. Figure 2 presents a comparison between an image sample in its original coloration and the same one after the conversion to grayscale.

Then, in a second approach, the image samples were pseudo-colored to create a new color pattern to highlight the EGC. The pseudo-coloring is a technique that tries to color a grayscale image sample. This is commonly achieved by mapping a single grayscale value to an RGB value, which is usually referred to as color maps. Since the image samples used in this work were originally in the RGB color system, it was necessary to convert them to the grayscale to apply such a technique. A representation of a pseudo-coloring may be observed in Fig. 3.

It is worth mentioning that these kinds of operations do not affect the image samples' existing texture.

Most of the captured EGC image samples have a lack of sharpness in their edges/shapes. These images result from the immunohistochemical reaction for a protein expressed exclusively in EGC, the S10 protein. This protein can be irregularly distributed in the cell, which can form irregular outlines and, associated with the low resolution of the microscopy used, often generates images with a blurred aspect.

Considering that, the second goal aimed to reduce the existing blur in the image samples and increase the edges' definition. To achieve that, the data samples had their edges (or borders) highlighted, by detecting and adding them to the original image samples. This is made possible by edge detection methods, that generally apply a filter by using different kernels. In this work, three different filters were used: Laplacian, Sobel, and Scharr. Figure 4 presents a comparison between examples generated by using these filters.

2.1.2 Texture descriptors

Different methods may be used to extract features from a given image. Regarding the visual attribute captured by these descriptors, we may observe that texture descriptors can potentially obtain good results in several different situations. And it stands out in different scenarios/applications, including in medical image analysis, biometric identification, etc. [26].

The texture of a digital image is characterized by variations in the color intensity. By observing the differences between the pixels of the images, it is possible to provide a practical way to analyze an object's texture. Such analysis overlaps other ways to make an image's interpretation, e.g., by color.

Among the different works described in the literature, carried out on diverse application domains, using texture-based features extraction approaches, we can mention: music genre classification [8], bird species classification [28], north atlantic right whale identification [14], identification of infants' cry motivation [12], speech recognition [36], acoustic scene classification [13], and COVID-19 identification using chest X-ray images [38], among others.

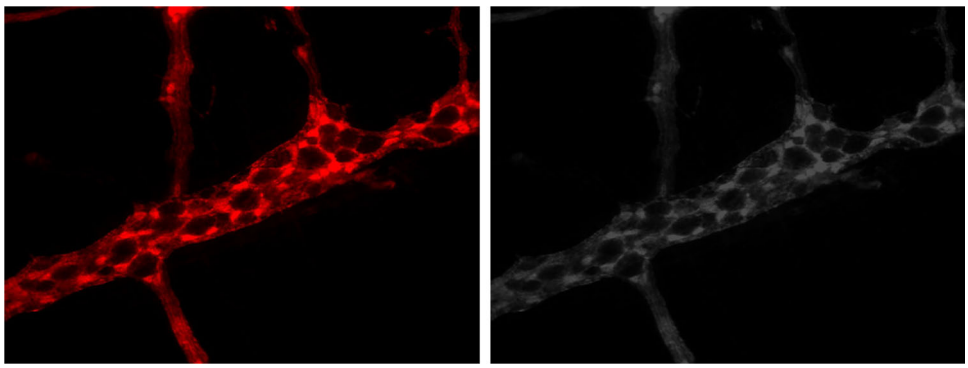

Fig. 2 Digital image sample of the AIA dataset (left) converted to the grayscale (right)

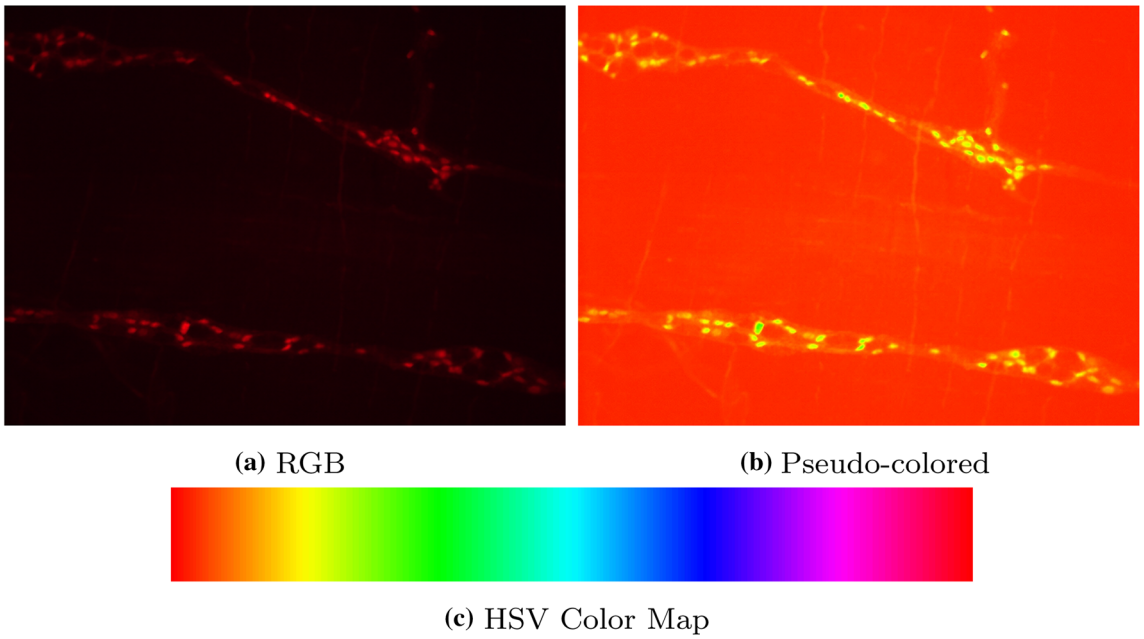

Fig. 3 Digital image sample of the D disease group (**a**) and the same one pseudo-colored (**b**) using the HSV color map (**c**)

In this work, three handcrafted texture descriptor approaches were used: Local Binary Pattern (LBP), Robust Local Binary Pattern (RLBP), and Local Phase Quantization (LPQ).

The LBP was originally proposed by Ojala et al. [31] and uses a local neighborhood from every input pixel to generate a representative binary value. Two main parameters may be cited: P and R. The P parameter represents the number of neighbor pixels from a central pixel c, while R represents the distance from it. The most common setup for LBP uses eight local neighbors ($P = 8$) two pixels distant from c ($R = 2$)[5]. The representation of such configuration can be described as $LBP_{8,2}$ or $LBP(8, 2)$. An extension of the original LBP defines the final feature vector as the normalized histogram that counts all uniform binary patterns (a binary pattern is considered uniform if it has no more than two transitions from 1 to 0 and vice-versa when evaluated as a circular list). This one has a total length of 59 features and presents better results when compared to the histogram of all individual binary patterns [5, 7, 31]. More information about the LBP can be found in [31].

The RLBP was originally proposed by Zhao et al. [52] as a variation of the LBP texture descriptor. According to the authors, the RLBP is more accurate to capture the textural content from images that contain noise interference, when it results in non-monotonic gray-level changes (even when such changes are not significant). The RLBP searches for a bit in the LBP pattern, that possibly suffered a variation inflicted by some kind of noise and then, review it. The original LBP's robustness is increased by this method, turning the binary patterns' uniformity concept a bit more flexible[5]. More details about the RLBP may be found in [5].

The LPQ was initially proposed by Ojansivu and Heikkil*ä* [32], being designed to be a texture descriptor more sensible to the image samples affected by blur. This

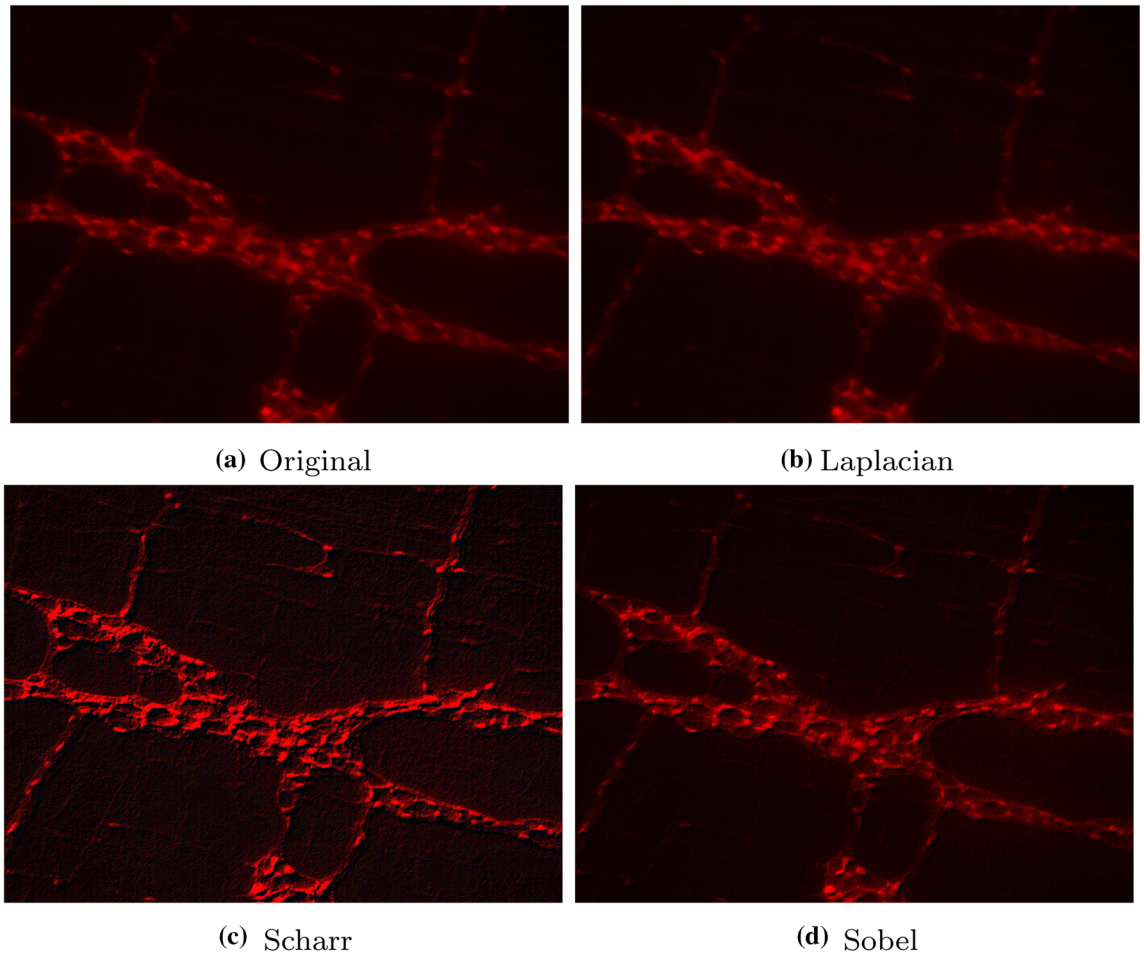

Fig. 4 Image sample from the C class, from the TW dataset. Sub-figure (**a**) presents the original data sample, while the remaining sub-figures show the resulting images by applying the edge highlighting methods using the following filters: Laplacian (**b**), Scharr (**c**) and Sobel (**d**)

descriptor has been presenting good performances, even in classification tasks, not targeted to images affected by this type of interference [7]. The LPQ uses periodic information from a bi-dimensional Discrete Fourier Transform (DFT), or specifically, a Short Term Fourier Transform (STFT). The STFT is computed in a rectangular neighborhood N_x for each pixel in an image sample. The rectangular window size (N_x) is an important parameter to be varied, considering its direct impact in the generated features. After locally computing the texture of each pixel, the resulting codes are presented in a histogram, similarly as the LBP method. More detailed information about the LPQ texture descriptor can be obtained in [32].

2.2 Classifier algorithms

In the machine learning context, the supervised learning task builds a hypothesis capable of predicting an unobserved data sample label, based on the knowledge obtained from a known dataset, containing labeled data samples. In other words, given a dataset with n examples of input/output pairs (x_1, y_1), (x_2, y_2), ..., (x_n, y_n), in which x and y may represent any value (not necessarily numerical), it is built an hypothesis (function h) that approximates to a true function f that generates each y_i value starting from the x_i value, i.e., $y = f(x)$. To evaluate this hypothesis, a set of test data samples is used.

This kind of task can be divided into two categories: regression problems and classification problems. If the output y assumes a finite set of values, the task is categorized as a classification problem. Otherwise, if y assumes a continuous numerical value the task is categorized as a regression problem [42].

To the accomplishment of classification tasks, classification algorithms may be used. Different classification algorithms are described in the literature. In this work, four of them were approached: Support Vector Machines (SVM), Gradient Boosting (GB), Random Forests (RF), Naïve Bayes (NB) and k-Nearest Neighbors (k-NN).

The SVM is a well-known method widely used by its efficiency to perform classification tasks. In its training step, a hyperplane is built, i.e., a decision limit with the shortest distance between the example points. In other words, it searches for a line (or surface) that segregates the patterns from different classes, being the margin defined as the distance from this one to the closest pattern. The support vectors are the transformed patterns that delimit such margin. The input is mapped to a higher dimensional space by a nonlinear function, using what is known as Kernel Trick. Such action is performed based on the fact that some data may not be linearly separable in its original input space, but being easily separable when another dimension is added [11, 42].

The NB can be categorized as a Bayesian Classification Algorithm. Such a category of algorithms identify an object based on the posterior probability. Thereby, an object's class is assumed by the Bayes's theorem. Introduced originally by Thomas Bayes (1702-1761), the Bayes's Theorem (or the Bayes's Rule) is a simple equation, quiet often used by most modern artificial intelligence systems as a base for probabilistic inferences. It allows the calculation of a new unknown probability, by using three giving known conditional probabilities, that can usually be easily found. Based on this logic, the NB classifier assumes that a dataset's attributes are conditionally independent of each other. The prediction of an unseen data sample is then performed, based on the probabilities calculated from its attribute values, giving the labeled data samples, and targeting one specific class [42].

The RF algorithm was originally proposed as a method of building classifiers based on decision trees, being as capable of increasing the accuracy in the training step as for samples not previously observed. Its operation can be shortly described as the build of multiple decision trees in a randomly selected subspace inside the features space. Then, generalizing the classification in different complementary approaches. By the end of the method's process, the tree-structured classifiers $h(x, \theta i), i \in \{1, .., k\}$, being k equivalent to the total number of decision trees, cast individual votes for one of the possible classes of x. The final prediction is assumed to be the most popular class, i. e. the class with the greatest number of votes. More details about this algorithm may be found in [19].

The K-NN is a instance-based learning algorithm and has its operation based on the nearest neighbor rule. Considering a dataset with n labeled samples $D^n = x_1, ..., x_n$, a certain x' sample, such that $x' \in D^n$, can be described as being the closest point to a test sample x. By using the nearest neighbor rule, x is classified with the same label/-class as x'. This rule can be naturally extended to operate with a larger number of neighbors. In this way, k nearest neighbors to x are used to perform the decision when classifying such a test sample. In other words, each of the neighbors cast a vote for one of the possible classes, and at the end, x is classified with the most popular class, i. e. the class with the largest number of votes. Considering this context, the traditional nearest neighbor rule is assumed as having $k = 1$. It is noteworthy that to avoid draws, the value of k always assumes an odd integer [11, 42].

Boosting refers to a general and effective method of producing an accurate/efficient learner by combining a set of weak (or moderately inaccurate) learners, [43]. Many methods have been developed based on such concept, e.g., Gradient Boosting, AdaBoost, and others. In this work, the GB algorithm is adopted. The algorithm's core functionality is based on constructing the new base-learners to be maximally correlated with the negative gradient of the loss function, associated with the whole ensemble [30]. So, the final model can reduce the error over time, considering the errors made by the previous predictors.

2.3 Non-handcrafted modeling

Traditionally, in the pattern recognition framework, features are extracted from the dataset and used as input in machine learning algorithms that are supposed to learn how to discriminate patterns from different classes. Thereby, a significant part of the effort put in works based on machine learning algorithms is dedicated to feature engineering. This is a time-consuming and challenging process that requires specialized knowledge.

That being said, we introduce the Feature Learning (FL), or Representation Learning (RL), concept. In this category, the techniques can automatically generate data representations, favoring the extraction of useful information during the build of classifiers and other predictive systems [1]. Among the countless ways to perform FL, Deep Learning methods, such as Convolutional Neural Networks (CNN), can be highlighted. Keeping that in mind, this section describes the non-handcrafted modeling, which aims to classify features extracted automatically by FL, being here referred to as non-handcrafted features. Figure 5 illustrates the general scheme employed to obtain features from the penultimate layer of a CNN.

2.3.1 Pre-processing

Different approaches can be described to avoid overfitting, helping to ensure better and most accurate classification rates. One of them, Data Augmentation, artificially increases the total number of image samples by performing small changes in the original samples, using different operations [47].

This approach is usually employed as a pre-processing technique when working with CNNs, taking into account

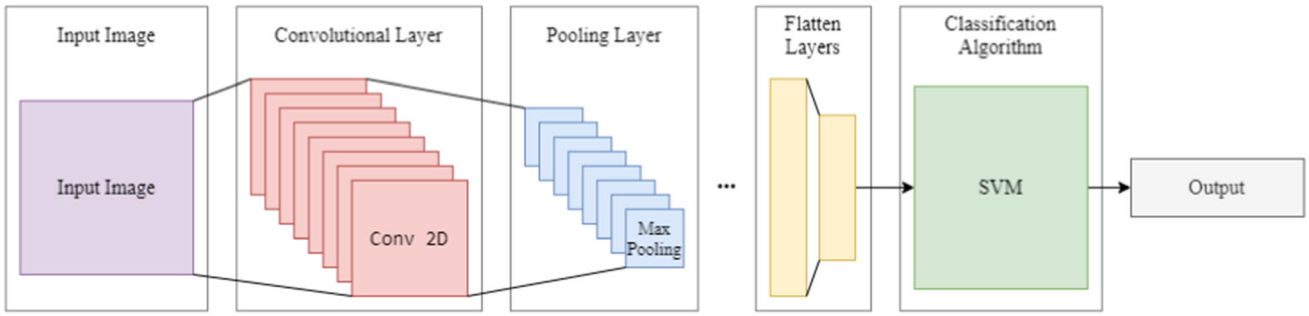

Fig. 5 General scheme of the feature learning

the huge amount of data required to achieve satisfactory classification rates. Such a technique was applied in this work, considering the modest quantity of image samples existing in the used datasets. At this point, it is important to observe that the limited size of the dataset used in this work is virtually insurmountable. It involves the induction of diseases to the research animals. It also needs to be approved in a long-term protocol by the "Standing Committee on Ethics in Animals Experimentation" of our university.

To generate new image samples based on the ones available, randomly chosen image processing operations were applied to the original samples, such as: adding Gaussian noises, contrast normalization, adding blur noises, vertical/horizontal rotations, saturation variations, sharpen and others. Some examples of generated image samples can be seen in Fig. 6.

2.3.2 Convolution neural networks (CNN)

CNN is an efficient algorithm, widely used in pattern recognition and image processing areas, due to its favorable characteristics, such as simple structure, less training parameters, and adaptability. Its shared weights structure makes it more similar to the neurons' connectivity pattern found in the human brain. Been heavily inspired in the human visual system structure, each neuron from a CNN does not globally visualize an entire image sample. But instead, only a portion of it (a local area) is visualized. This way, reducing the network model's complexity and the number of weights [24, 25].

A CNN can be considered a variation of the Multi-Layer Perceptron networks, been capable of applying filters in visual data, keeping the neighborhood relation between the image pixels through the network processing [50].

Traditionally, the CNN's layers can be segregated in two sets: (1) the set of layers responsible for the image sample's feature extraction by FL (in this work, named non-handcrafted features), is usually composed of convolutional and pooling layers; and (2) the set of layers responsible for the classification, being composed of one or more fully connected (dense) layers [27, 50]. It is worth mentioning that the total number of layers may differ from one CNN network architecture to another.

2.4 Transfer learning

Traditionally, machine learning algorithms predict the label of unknown data samples using models that were previously trained in a labeled dataset. With the evolution of such algorithms and the complexity growth of the tasks that employ them, it creates the need to heave an enormous amount of training data to achieve satisfactory results [33].

The knowledge transferring concept induces the thought that a previous knowledge, acquired for a specific purpose, may be reused in another one. Having that in mind, the Transfer Learning ideal is presented. It aims to resolve new problems faster and with better solutions by using pre-obtained knowledge [33].

Transfer Learning extracts knowledge from a source task and uses it in a target task. When using CNNs, this method makes possible the use of pre-trained models (trained in different datasets/purposes) in new classification tasks. Thus, demanding a smaller number of data samples in CNN's training. Once the FL layers are kept unchanged, only requiring the classifications layers to be trained. It is worth mentioning that the quantity and dimensions of such layers may differ from those used originally in the base model, aiming to adapt the network to the target task. Also, it can be replaced by traditional classification algorithms, e.g., Support Vector Machines (SVM).

Finally, it is essential to emphasize that transfer learning in this work can be seen as a timely strategy to deal with issues regarding the dataset's modest size. As already pointed, in our specific application, it is not easy to enlarge the dataset since it needs new animal resources and the appliance of a long-term method.

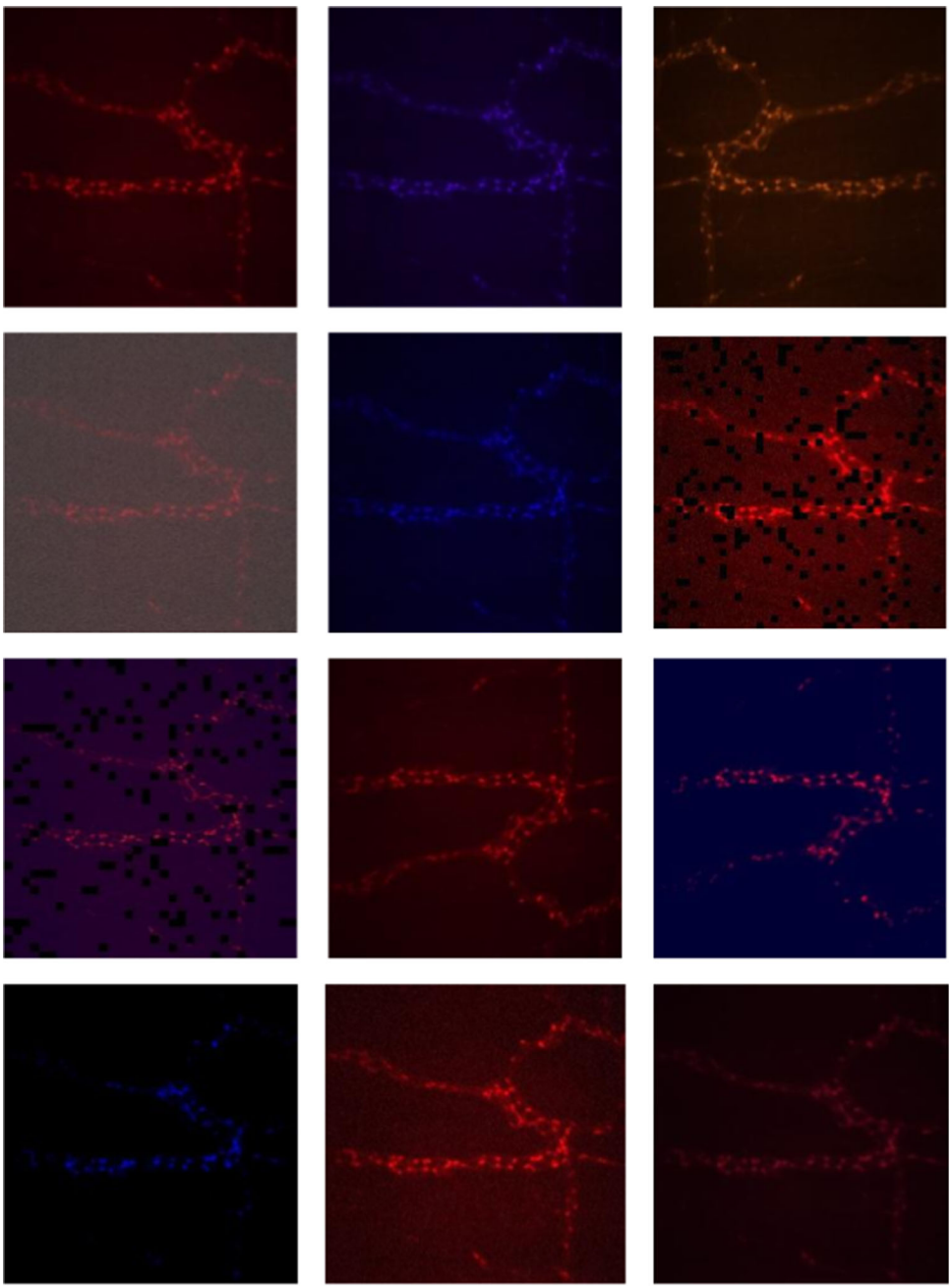

Fig. 6 Examples of samples generated using Data Augmentation, by applying four randomly chosen techniques. Some of the techniques experimented are: adding Gaussian noise and/or blur; adding hue and saturation; flipping the image vertically and/or horizontally; changing the color space to BRG; Coarse Dropout; and others. The original image sample can be seen in the top left corner

2.5 Classifier combination

Many classification algorithms used to generate probability estimates in their outputs. Such values reference the prediction scores for each class present in the problem, obtained from the evaluation of test samples. It is possible to perform different operations over these probability scores aiming to combine them. This way, new predictions scores are obtained based on the previously reached values, generating a final result.

Three classifiers combination rules, originally proposed by Kittler et al. [21], were used in this work for such finality. In these equations, x refers to the pattern to be classified, n is the number of classifiers involved in the combination, c is the number of classes, y_i is the output label of the i_{th} classifier in a problem with the following possibilities of classes labels $\Omega = \omega_1, \omega_2, ..., \omega_c$, and $P(\omega_k|y_i(x))$ is the probability of the sample x to belong to the class ω_k according to the i_{th} classifier.

– Max Rule: for each existing class from the dataset, the maximum value between the prediction scores from different classifiers is chosen. Later on, the final result is given by the class with the biggest score. This combination rule can be represented by Equation 1.

$$max(x) = \arg\max_{k=1}^{c} max_{i=1}^{n} P(\omega_k|y_i(x)) \quad (1)$$

– Sum Rule: considering all generated classifiers, the calculated predictions are summed for each class. Then, the class with the maximum score is chosen as the final result. This combination rule can be represented by Equation 2.

$$sum(x) = \arg\max_{k=1}^{c} \sum_{i=1}^{n} P(\omega_k|y_i(x)) \quad (2)$$

– Product Rule: this rule works similarly as the sum rule. But, instead of performing a sum operation, the values are multiplied. This combination rule can be represented by Equation 3.

$$prod(x) = \arg\max_{k=1}^{c} \prod_{i=1}^{n} P(\omega_k|y_i(x)) \quad (3)$$

It is important to remember that by combining classifiers, we have an opportunity to merge results obtained using handcrafted and non-handcrafted features once we have probabilities predictions as output from classifiers on both modes. The combination of handcrafted and non-handcrafted approaches aiming to get a single final decision has somehow already proven to be effective in other works [9, 29].

3 The datasets

This work presents three novel datasets created by researchers of the Enteric Neural Plasticity Laboratory of the State University of Maringá. Such datasets are composed of image samples obtained from the ENS of rats, in which EGC can be visualized through the immunostaining of the S100 protein. Figure 7 shows some image samples taken from the datasets. Each dataset represents an investigated disease, being them: arthritis rheumatoid (AIA), cancer (TW) and diabetes mellitus (D).

Each dataset is represented by the diseases' name abbreviations (AIA, TW, and D) in this work. It is worth mentioning that in this scenario, the datasets can be also called "disease groups." The datasets are composed of two classes, one containing image samples extracted from sick animals (S) and other from control/healthy (C) ones. The exact quantity of image samples per dataset and per class, and the image samples' dimensions, can be seen in Table 1.

The datasets were created taking into account the ethical principles under the terms set out in the Brazilian federal law[1], established by the Brazilian Society of Science on Laboratory Animals (SBCAL). All the proceedings were submitted and approved by the Standing Committee on Ethics in Animals Experimentation of the State University of Maringá[2]. After the experimental time, the animals were frozen and sent to incineration.

The jejunum, i.e., the second part of the small intestine, of male adult Wistar rats (Rattus norvegicus), albius variety (D and TW) and holtzmann rats (AIA) were used in this study. The experimental models were developed according to the works of Frez et al. (2017) [15] (D), Vicentini et al. (2017) [51] (TW) and Souza et al. (2011) [46] (AIA). Variations in the animals' ages were encountered as well, in every disease investigated here. The animals' euthanasia was performed with tiopental (150 mg/kg)[3] intraperitoneally. Then, the jejunum was collected and processed to evidence the EGC.

The jejunum fixation and S100 immunostaining were performed according to the protocol proposed by Pereira et al. (2011) [39]. The images of the EGC were obtained using an optic microscope[4] with immunofluorescence filters and high-resolution camera[5] attached to a computer. Photomicrographs were recorded using Motic Images Plus 2.0ML software (Motic China Group Co.). The image samples were then obtained using 20×, from randomly chosen places of the animals' jejunum.

It is worth mentioning that the image samples' acquiring is variable according to the target dataset. It can be justified mainly by the possibility of distinct tissue fixations and specific immunostaining responses. More detailed information about these processes and the datasets, such as the induction of the disease, may be found in [15] (D), [51] (TW), and [46] (AIA).

The datasets used in this work were made freely available for research purposes[6], such a way that other researches can take benefit from it, and properly compare the results obtained using different techniques with those obtained here.

[1] Law 11,794 (October 2008) and the decree 66,689 (July 2009).

[2] Protocol numbers 062/2012 (TW), 4462180216 (AIA) and 073/2014 (D).

[3] Abbot Laboratory, Chicago, IL, USA

[4] Olympus BX 41, Tokyo, Japan

[5] Moticam® 2500 5.0 Mega Pixel - Motic China Group Co., Shanghai, China

[6] https://github.com/gustavozf/EGC_Z_dataset.

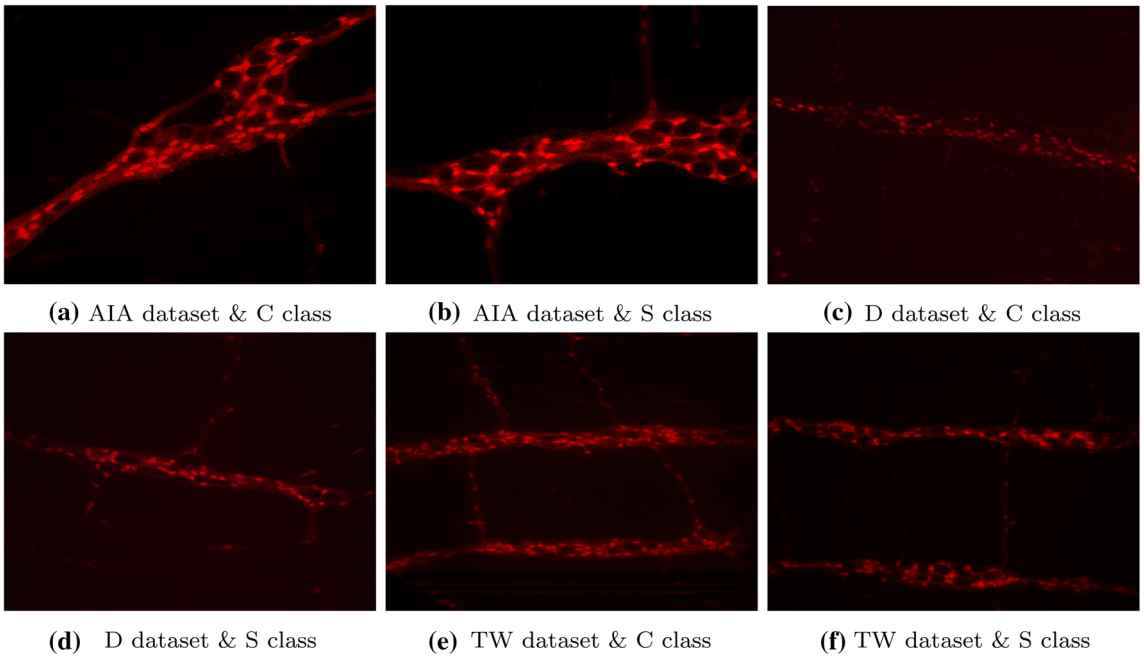

Fig. 7 Image samples from the datasets and both classes. The red coloration is resulted from the immunostaining of the S100 protein. The EGC can be visualized as the most bright points in the samples

Table 1 Diseases evaluated in this work, according to the categories' distribution in the experimental groups

Disease	Abbreviation	Image dimension	Number of samples	
			Sick (S)	Control (C)
Arthritis Rheumatoid	AIA	1024 × 768	210	208
Cancer (Walker's tumor-256)	TW	1384 × 1036	192	224
Diabetes Mellitus	D	1384 × 1036	290	224

4 Experimental analysis

In this section, we describe the results obtained after performing the experimental protocol designed in this work. Section 4.1 describes the exploratory investigation taken that leaded to the achievement of this work's best classification rates. Section 4.2 presents the parameters and implementations used in the algorithms employed in this work. Finally, Sect. 4.3 presents and briefly discusses this work's results.

4.1 Exploratory investigation

The experiments were performed, following a result-oriented exploratory research approach. Such experiments may be divided into three categories: (1) the ones that evaluated the performance of handcrafted features; (2) the ones that evaluated the performance of non-handcrafted features; and (3) the ones that combined the resulting classifiers from the previous categories.

4.1.1 Handcrafted modeling

The set of experiments using handcrafted features was designed to evaluate different approaches, such as assessing different texture descriptors, enhancing the image samples' visual features, and testing different classification algorithms.

The first experiments aimed to evaluate the performance of features extracted based on the image samples' texture. To accomplish such a goal, we used LBP, RLBP, and LPQ texture descriptors. These were applied to the image samples, and the resulting features were used as input in the SVM classification algorithm. It is worth mentioning that the assumed values of the following parameters: neighborhood size, distance from the central pixel (LBP and RLBP), and window size (LPQ), were chosen according to the best performances obtained by Costa [7].

From these experiments' results, it was possible to notice a tendency to find more significant results by using the LPQ texture descriptor and inducing that it would be

the most appropriate for the here approached problem. This may be justified by the fact that the LPQ was created to better extract features from image samples affected by blur, a kind of signal noise present in the image samples of this work's dataset. It is worth mentioning that the AIA's classification performance reached its best values by using the RLBP texture descriptor. That can be justified because the image samples from this disease group are less affected by blur and the ones more affected by the inconvenience of different noises generated by the immunohistochemistry background.

Having that in mind, new experiments were performed to reach better classification rates. To achieve this goal, new features were extracted from the image samples, using the LPQ texture descriptor and making variations the window size value, assuming the values 3, 5, 9, 11, and 13. From these results, it is possible to observe better classification rates than the previously performed experiments. Showing that there is still a space for improvement of the LPQ performance on this task if the parameters are properly adjusted.

From this point, we aimed to evaluate the performance of experiments carried out by applying pre-processing techniques. Firstly, the image samples had their colors omitted by converting them to grayscale to achieve this goal. Since these kinds of operations do not affect the existing texture in the image samples, the texture descriptors, and their parameters, used in these experiments were the ones that reached the best classification rates until this moment. The resulting F-measure values found from these experiments showed an improvement on the AIA and D datasets.

In a second moment, the image samples were pseudo-colored. Different color maps were evaluated to perform the pseudo-coloring, such as HSV, Autumn, Jet, Rainbow, and others. The resulting images were observed, aiming to search for the best choice that created greater visibility of the EGC. From the ones tested, it was possible to note that the HSV colormap best suited the searched goal.

By analyzing the results obtained from this experiment, we observe an opposite behavior compared to the results that converted the image samples to the grayscale. For the AIA and TW datasets, the found F-Measure values were inferior to the ones found in past experiments. While for the D disease group, a slight improvement could be observed compared to the best one found until this point.

Then, considering the existing blur in the image samples, new experiments were performed aiming to reduce this kind of noise and increasing the definition of the edges. The data samples had their edges (or borders) highlighted to achieve this goal by detecting (with the Laplacian, Sobel, and Scharr filters) and adding them to the original image samples. For this approach, the LBP, RLBP, and LPQ texture descriptors were used to extract handcrafted features.

The choice for re-testing the LBP and RLBP texture descriptors in these experiments, even knowing the superior performance of the LPQ in this problem, is justified by the direct impact that the edge highlighting method generates in the image's texture. This creates the need to reevaluate the performance of these texture descriptors, giving the new environment to be explored. Besides, in this case, the test is performed with images supposedly free from undesirable effects of blurring (a characteristic that the LPQ excels at). The operations were applied in the entire dataset. From the new image samples generated, features were extracted using the texture descriptors previously mentioned and classified by the SVM classification algorithm.

By the end of these experiments, it is possible to conclude that although the image samples generated by edge highlighting methods may be visually better when used as input in classification systems, the values obtained from the classification rates were inferior to those obtained without such process. One may conclude that such an approach would be inefficient for the problem investigated here.

The experiments performed until this point aimed to identify the best possible way to represent the data samples of the approached problem, used as input through handcrafted features. In such experiments, the SVM classification algorithm was always used. Therefore, we carried out new experiments to evaluate different classification algorithms' performance, using best-extracted features found, i.e., features that led to the best classification rates until now, as input to the algorithms. To achieve this goal, the k-NN, GB, RF, and NB classification algorithms were tested. However, the results found during these experiments' execution were still inferior to the ones found using the SVM classification algorithm.

4.1.2 Non-handcrafted modeling

From this point, the methods that use non-handcrafted features, i.e., features extracted automatically by FL, are evaluated. Therefore, experiments were performed using different CNN architectures and transfer learning.

The CNN tests were performed using the LeNet5, AlexNet, and MaxNet CNN architectures. From these experiments, it was possible to conclude that, although the number of samples in the dataset was artificially increased, the experimental models' training was not able to succeed in the task of abstracting the problem in question.

New experiments were them employed, aiming to investigate the efficiency of applying the Transfer Learning technique in the problem investigated here. At the end of the proposed experiments, it is possible to notice good

performance rates obtained by applying the Transfer Learning technique. These rates reached equivalent or slightly greater values when compared to the ones resulted by classifying handcrafted features. Showing that such a technique would be more appropriate for the classification of two out of three datasets addressed here, AIA and D. Justified by the fact that the strategy accomplished here does not demand a huge amount of data samples, like traditional CNN classification methods do, and extracts features using pre-trained FL layers proposed to solve problems from different domains.

4.1.3 Classifiers combination

Previously performed experiments used handcrafted and non-handcrafted features as input to create classification models. By such experiments, it was possible to observe the performance of different feature extraction and classification techniques to ascertain the generated classifications rates' behaviors on the investigated problem. With this in mind, the last experiments performed aimed to combine the most promising classifiers presented until this point for each scenario, i.e., handcrafted and non-handcrafted.

For each dataset, six classifiers were chosen, three of them obtained from handcrafted features, and the other three from non-handcrafted features. Thus, all possible sets of configurations were generated, which is equivalent to 57 different possibilities ($2^6 - 7 = 57$, ignoring the empty set and the ones with length equal to one) for each disease group. These were combined using the Sum, Product, and Max rules. At the end of these experiments, one may notice an improvement in the classification rates for every disease group, which indicates a complementarity between the classifiers (handcrafted and non-handcrafted) used.

4.2 Parameters and configurations

The image samples were pseudo-colored using the colormaps made available by the OpenCV library[7]. The edge enhancement and the image samples' conversion to the grayscale were performed using such a library.

The SVM implementation used here belongs to the libSVM library [3]. We have used the Radial Basis Function (RBF) kernel, and a grid search procedure optimized the parameters C and γ. The remaining classification algorithms implementation used here are those available in the Scikit-learn [37] library. The k-NN algorithm used five nearest neighbors ($k = 5$) to perform the classification. The distance was calculated by the Minkowski Distance with $p = 2$ and the voting system had uniform weights, i.e., the neighbors' votes had the same weight in the final prediction. The RF algorithm used ten trees, and the Gaussian version of the NB algorithm was used. The GB algorithm optimized the deviance loss function, with a learning rate equivalent to 0.1, 100 estimators and quality of the split measured by the mean squared error with an improvement score by Friedman. Such parameters were chosen considering the default values suggested by the *scikit-learn* library.

The CNNs used were implemented by the use of the Keras [6] library. The created models were compiled using the Adam Optimizer [20], with the β_1, β_2 and decay parameters equivalent to 0.9, 0.999 and 0.0005 respectively.

The CNN tests were performed using the LeNet5, AlexNet, and MaxNet CNN architectures, with batch size equal to 128 and learning rate varying between 10^{-3} and 10^{-5}. The number of epochs was set to 1024. It is worth mentioning that the training process approached two main callback functions: Early Stopping and Model Checkpoint.

The Model Checkpoint function was used during the training process to save the best model during the training iterations. It monitored the accuracy value obtained by the current i^{th} iteration and saved it once such value was greater than those found on prior iterations. After the training step, the best saved model was used for testing.

The other callback function, Early Stopping, was employed to avoid the execution of an unnecessary amount of epochs while training the models. As early experiments were performed, it was possible to notice that the models converged in a relatively small quantity of epochs. Having that in mind, such a function was set to monitor the validation loss value, stopping the training process if it did not improve in the last 64 epochs. Such protocol was repeated for each of the k-fold cross-validation rounds (with k=10).

To reduce the possibility of overfitting in the CNNs that did not approached Transfer Learning, two different techniques were applied: (1) artificially increasing the number of image samples through data augmentation, generating 32 new image samples from each of the existing ones; and (2) using dropout layers in the CNN architectures.

The Transfer Learning experiments used the FL layers of the VGG16, InceptionV3, and InveptionResNetV2 architectures to extract features of the EGC images. It is worth mentioning that these layers' weights were obtained by performing the training in the ImageNet dataset. The implementations of such networks are all made available by the Keras library. Unlike common approaches, instead of such features being classified by a new set of fully-connected (dense) classification layers, the SVM classification algorithm was used for such a task. This is justified by the results previously described in this study.

[7] www.opencv.org.

Additionally, considering that the FL layers of the CNN architectures that approached the Transfer Learning technique, generate a great number of features, the Chi-square (χ^2) statistical test was employed as a Feature Selection technique, aiming to reduce the number of features to a smaller value n. By doing so, computational time would be reduced without fully compromising the found results. To accomplish this goal, the aforementioned method was applied after the features were extracted by the pre-trained feature learning layers of the CNN models. The selected n resulting features for each image sample were then used as input for the classification algorithm. Being the tested values for n: 256, 512, 1024, 2048, and 4096 features.

4.3 Results

This section presents the classification rates found when executing the experiments accordingly to the exploratory investigation presented in Sect. 4.1.

4.3.1 Handcrafted modeling

Table 2 presents the results found in the experiments that aimed to evaluate the performance of features extracted by texture descriptors. It can be observed that for the AIA disease, the best results[8] were found when the RLBP texture descriptor was used, with a F-Measure of 0.8062. For the remaining diseases, such results were found by using the LPQ texture descriptor, reaching F-Measures of 0.9447 and 0.8637 for TW and D diseases, respectively.

Table 3 reports the experiments' classification rates that extracted features using the LPQ texture descriptor and used variations in the window size value. By analyzing these experiments, it was possible to observe a better efficiency of features extracted by the LPQ texture descriptor. Table 4 compares the best results.

In Table 4, it is possible to observe F-measure values of 0.8110, 0.9712, and 0.8773, respectively, for the AIA, TW, and D datasets. All of the recently mentioned values were reached by classifying handcrafted features extracted by the LPQ texture descriptor.

The results for the experiments that classified the image samples converted to the grayscale are in Table 5. In this case, one may observe an improvement in the classification rates in two of the three approached disease groups, being them: AIA and D. In both of them, the F-Measure values reached an average improvement of 2 percentage points, presenting 0.8325 and 0.8907, respectively. For the TW disease group, the greatest found F-Measure value was equivalent to 0.9471, being 0.0241 inferior to the highest value found in past experiments.

Table 2 F-Measure values found from the experiments using different texture descriptors

Dataset	Texture Descriptor		
	$LBP_{8,2}$	$RLBP_{8,2}$	LPQ_7
AIA	0.7895	**0.8062**	0.7535
TW	0.9279	0.9231	**0.9447**
D	0.8520	0.8327	**0.8637**

Based on these values, it is possible to observe that the best classification rates found in this work until this point, were reached with the LPQ texture descriptor for all the evaluated diseases. With the window size parameter being equivalent to 11 for AIA, and 13 for TW and D.

The results of the experiments that pseudo-colored the image samples are presented in Table 6. When analyzing them, it can be observed that for the D dataset, a slight difference of 0.006 can be noticed, when compared to the best one found, described in Table 5.

By the end of the experiments that varied the image samples' coloration, it is possible to compare the variations applied with the image samples' on the original color system (RGB) and analyze the overall performance of them. Figure 8 presents a visual comparison between all of the best results obtained for each of the color variations applied in the image samples. It is possible to observe that a different variation of the image sample was used to achieve its best F-Measure value for each disease group.

For the AIA disease group, the image samples that were converted to grayscale reached a better F-Measure than other approaches. Its value was equivalent to 0.8325, nearly 2 percentage points more than the remaining color systems. To achieve this value, the LPQ_{11} texture descriptor was used.

The TW disease group, by its original color system (RGB), presented a performance of approximately 2.5 percentage points greater when compared to the other two. The best F-Measure was equivalent to 0.9712 and it was obtained by the classification of handcrafted features extracted using the LPQ_{13} texture descriptor.

Finally, the D disease group achieved its best F-Measure value, equivalent to 0.8967, by the extraction of handcrafted features through the use of the LPQ_{13} texture descriptor and using the image samples pseudo-colored to the HSV colormap.

The F-Measure values obtained from the experiments that highlighted the image samples' edges are shown in Table 7. Figure 9 presents a graphical comparison between

[8] In this work, the best results will always be defined by the best F-measure performance, once this metric is given by the harmonic mean between recall and precision.

Table 3 F-measure values found from experiments that explored different values for the LPQ's window size

Dataset	Window Size				
	3	5	9	11	13
AIA	0.7751	0.7679	0.7943	**0.8110**	0.8014
TW	0.9447	0.9591	0.9447	0.9567	**0.9712**
D	0.7838	0.8189	**0.8773**	0.8690	0.8636

Table 4 Best results found in the experiments that evaluated the performance of handcrafted features, extracted by the use of different texture descriptors

Dataset	Texture descriptor	F-measure
	LPQ_{13}	0.8014
AIA	LPQ_{11}	**0.8110**
	$RLBP_{8,2}$	0.8062
	LPQ_{13}	**0.9712**
TW	LPQ_{11}	0.9567
	LPQ_{5}	0.9591
	LPQ_{13}	0.8636
D	LPQ_{11}	0.8690
	LPQ_{9}	**0.8773**

Table 5 Classification rates found by the tests where handcrafted features were extracted from the image samples converted to the grayscale

Dataset	Texture descriptor	F-measure
	LPQ_{13}	0.8038
AIA	LPQ_{11}	**0.8325**
	$RLBP_{8,2}$	0.7775
	LPQ_{13}	**0.9471**
TW	LPQ_{11}	0.9254
	LPQ_{5}	0.9303
	LPQ_{13}	**0.8907**
D	LPQ_{11}	0.8851
	LPQ_{9}	0.8696

the best results found in this tests and the ones previously obtained.

For the AIA dataset, it is possible to observe that the Scharr filter had the best results. By the use of such a filter and the LBP texture descriptor, it was reached a F-Measure value of 0.8277. Compared to the best values found by the use of each filter, the recently mentioned value performed approximately 2 percentage points better.

For the TW dataset, the best F-Measure value represented 0.9471, and it was reached using the RLBP texture descriptor and the Sobel Filter. It is worth mentioning that the values found in the classification rates for this dataset showed a high similarity level. These, with few exceptions, were kept around 0.94.

Finally, for the D dataset, the highest F-Measure value obtained was equivalent to 0.8695. It was reached by features extracted with the LPQ texture descriptor and edges highlighted by the Laplacian filter. Comparing the F-Measure values obtained in these tests, it is possible to observe a higher performance when the LPQ texture descriptor was used. That shows its potential in the scenario when the image samples are less affected by blur.

The next experiments aimed to evaluate the performance of different classification algorithms. The features used as input in the experiments carried out here were extracted by:

- **AIA:** the LPQ_{11} texture descriptor, with the image samples converted to the grayscale;
- **TW:** the LPQ_{13} texture descriptor, with the image samples in their original coloration;
- **D:** the LPQ_{13} texture descriptor, with the image samples pseudo-colored to the HSV colormap.

The results obtained from the execution of these tests can be seen in Table 8. In this one, it is possible to observe that the RF classification algorithm resulted in better classification rates than the other ones. In contrast, the NB algorithm obtained the lowest classification rates in the executed tests. Compared to the ones obtained by the RF algorithm, the NB's F-Measure values were averagely 0.0837 inferior. Such results may be justified by the algorithm's nature, considering that its performance stands out usually in smaller datasets with categorical features.

A comparison can be seen represented in Fig. 10. In this one, it is possible to observe the SVM algorithm's superior performance in the same set of features, having its values averagely 0.0638 greater when compared to the values achieved by the RF algorithm, which had the best classification rates in the tests here executed.

4.3.2 Non-handcrafted modeling

From this moment, we describe the experiments carried out using CNNs to extract the features and perform the classification for the proposed problem. The best-reached results by each used architecture can be seen in Table 9.

In Table 9, it is possible to observe that for the set of image samples of the AIA disease group, the LeNet5 CNN architecture presented the best F-Measure value. This leads

Table 6 Results obtained from the tests where handcrafted features were extracted from image samples pseudo-colored with the HSV colormap

Dataset	Texture descriptor	F-measure
	LPQ_{13}	0.7990
AIA	LPQ_{11}	**0.8134**
	$RLBP_{8,2}$	0.7942
	LPQ_{13}	**0.9495**
TW	LPQ_{11}	0.9327
	LPQ_{5}	0.9183
	LPQ_{13}	**0.8967**
D	LPQ_{11}	0.8811
	LPQ_{9}	0.8927

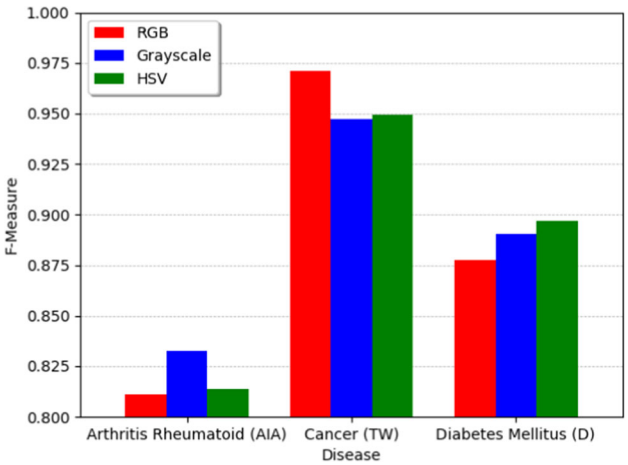

Fig. 8 Comparison between the best results obtained from the classifications that aimed to make variations in the image samples' coloration

Table 7 F-Measure values found from the experiment that highlighted the edges of the image samples, through the use of the Laplacian, Sobel, and Scharr filters

Disease	Texture descriptor	Filter		
		Laplacian	Scharr	Sobel
	$LBP_{8,2}$	0.7942	**0.8277**	0.8062
AIA	LPQ_{11}	0.7990	0.8158	0.8062
	$RLBP_{8,2}$	0.8038	0.8253	0.7918
	$LBP_{8,2}$	0.8992	0.9327	0.9423
TW	LPQ_{13}	0.9424	0.9448	0.9423
	$RLBP_{8,2}$	0.9279	0.9448	**0.9471**
	$LBP_{8,2}$	0.8442	0.8105	0.8171
D	LPQ_{13}	**0.8695**	0.8304	0.8675
	$RLBP_{8,2}$	0.8129	0.7991	0.8128

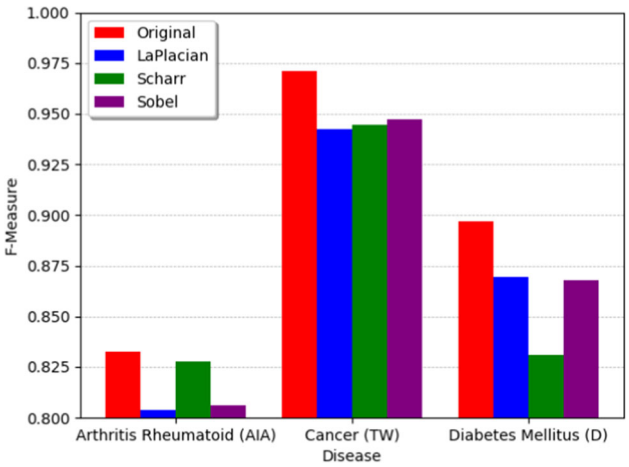

Fig. 9 Comparison between the resulting F-Measure values obtained from the classifications where the image samples had their shapes/edges highlighted and the ones obtained from the unchanged image samples (original)

us to conclude that a simpler architecture would be satisfactory for this work's goal in this scenario. The TW and D disease groups' set of image samples otherwise had their best F-Measure values found by using the AlexNet architecture. This one may be considered as a more robust architecture. However, when these results are compared to the ones found through handcrafted features, it is possible to observe lower classification rates, having a difference in the F-Measure values of 0.1194 (average).

The best results from executing the experiments that employed the Transfer Learning technique may be found in Table 10. We may notice that the InceptionResNetV2 architecture achieved the worst F-Measure values, around 22 (average) percentage points inferior to the best values found in these experiments.

For the AIA disease group, the best results were reached using the InceptionV3 architecture and 4096 features. In this case, the F-Measure value was 0.8468. The TW and D disease groups reached their best results using the VGG16 architecture, with the feature quantity reduced to 2048 (TW) and 4096 (D). It is worth mentioning that the best

Table 8 Classification rates obtained from the experiments that tested the execution of different classification algorithms. The results found by the classification using the SVM algorithm are also shown for comparison reasons

Dataset	Classification algorithm				
	k-NN	RF	NB	GB	SVM
AIA	0.7249	**0.7847**	0.7215	0.7799	0.8325
TW	0.8776	**0.8920**	0.7672	0.8968	0.9712
D	0.8286	**0.8325**	0.7694	0.8576	0.8967

F-Measure values for the AIA and D datasets surpassed the best ones found until this point, obtained using handcrafted-features, by 0.0143 and 0.0065, respectively.

4.3.3 Classifiers combination

Finally, this last experiment evaluates the impacts of combining classifiers. For this purpose, we selected the most promising classifiers presented until this point. Table 11 briefly introduces them. It is worth mentioning that although the classifiers that approached the GB algorithm presented better results when compared to the RF algorithm in an isolated way, this second one was chosen to perform the combinations shown here due to its more efficient observed complementary.

The best results obtained by the combination are reported in Table 12.

For the AIA dataset, the combination of three classifiers, through the max rule, produced a F-measure value of 0.893, which is 4.62 percentage points better than the best one found so far (available in Table 10). Among the classifiers used, one used handcrafted features, while the other two used non-handcrafted features. The classifiers with IDs equivalent to 3 and 4 were extremely influential to the classification rates obtained in these tests. Both of them are present in all of the best results found.

When analyzing the TW dataset classification rates, it is possible to observe that the best F-Measure value found was equivalent to 0.9845. Approximately 1.33 percentage points more than the best F-Measure value found until this point (observable in Table 3) for such dataset. In total, five classifiers were combined by the sum rule to achieve these values. The combination counted with three classifiers that used handcrafted features and two that used non-handcrafted features.

For the D dataset, the best F-measure value was 0.9513, which is 5.13 percentage points better than the best value found for this dataset (available in Table 10). This result was possible by combining three classifiers using the sum rule. In this case, two classifiers used handcrafted features and one non-handcrafted. It is worth mentioning that the classifiers with ID 1 and 2 can be considered essential to obtaining these performances.

Figure 11 shows a graphical representation of the configurations used to combine the classifiers that achieved the best classification rates presented in this section, recently displayed in Table 12, for each dataset approached in this work.

Table 9 Best results obtained for each CNN architecture, from the experiments that used them to perform the FL and classification

Dataset	CNN architecture	Learning rate	F-Measure
AIA	LeNet5	10^{-3}	**0.7224**
	MaxNet	10^{-4}	0.6904
	AlexNet	10^{-5}	0.6984
TW	LeNet5	10^{-3}	0.8035
	MaxNet	10^{-3}	0.8199
	AlexNet	10^{-4}	**0.8721**
D	LeNet5	10^{-3}	0.6772
	MaxNet	10^{-4}	0.7216
	AlexNet	10^{-5}	**0.7475**

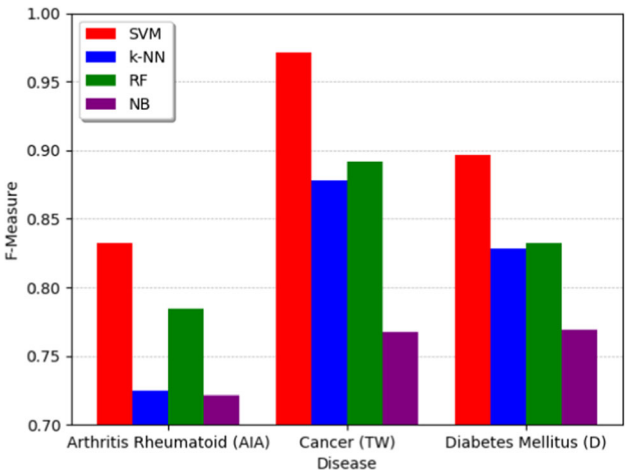

Fig. 10 Graphical comparison between the F-Measure values obtained by the classifications performed using different classification algorithms

Table 10 Best classification rates found from executing the tests using the Transfer Learning method, for each evaluated CNN architecture

Dataset	CNN	*n*	F-Measure
AIA	VGG16	4096	0.8421
	InceptionV3	4096	**0.8468**
	InceptionResNetV2	1024	0.6579
TW	VGG16	2048	**0.9327**
	InceptionV3	4096	0.8823
	InceptionResNetV2	2048	0.7444
D	VGG16	4096	**0.9043**
	InceptionV3	4096	0.7417
	InceptionResNetV2	2048	0.5982

5 Discussions

In this section, we discuss the results presented so far. It will be conducted by looking for answers to the following questions, aiming to get a more comprehensive understanding of the behavior of the methods and techniques experimented on the evaluated classification tasks:

- Which features provided the best results in each feature representation scenario and disease group?
- Which feature representation provided the best results?
- Which disease group is easier/harder to predict?
- Have the fusion strategies contributed to improving the results?

5.1 Which features provided the best results in each feature representation scenario and disease group?

This question is threefold, i.e., we have to consider the three different feature representation scenarios used in each dataset: the handcrafted features (LBP, RLBP, and LPQ); the non-handcrafted features with pre-configured CNNs (MaxNet, LeNet5, and AlexNet); and the non-handcrafted features with transfer learning (VGG16, InceptionV3, and InceptionResNetV2).

To answer the question, i.e., define the best features in each representation scenario/disease group, we used the statistical evaluation protocol proposed in Charte et al. (2015) [4]. Using this protocol, we calculate the ranking of the f-measures classification results obtained in all experiments with the different features based on the Friedman statistical test. The classification performances using the

Table 12 Best F-Measure values found from the experiments that combined classifiers

Dataset	Classifiers (IDs)	Type(s)*	Rule	F-Measure
AIA	3, 4	N and H	Max	0.8863
	3, 4, 5	N and H	Max	**0.8930**
	3, 4, 5	N and H	Sum	0.8899
	1, 3, 4, 5	N and H	Max	0.8858
	1, 3, 4, 6	N and H	Sum	0.8806
	1, 2, 5	N and H	Sum	0.9818
	1, 3, 4	N and H	Sum	0.9767
TW	1, 4, 5	N and H	Sum	0.9789
	1, 2, 3, 6	N and H	Product	0.9739
	1, 2, 3, 5, 6	N and H	Sum	**0.9845**
	1, 2	H	Max	0.9413
	1, 2, 4	N and H	Sum	**0.9513**
D	1, 2, 4	N and H	Product	0.9495
	1, 2, 4, 5	N and H	Sum	0.9459
	1, 2, 4, 6	N and H	Sum	0.9445

*Type(s) of classifiers involved in the combination.

**N stands for "non-handcrafted" features, and H stands for "handcrafted" features

Table 11 Description of the classifiers used to execute the experiments that employed classifiers combination techniques

Dataset	ID	Feature Extractor	Classification	Observations	F-Measure
AIA	1	LPQ_{11}	SVM	Samples converted to the greyscale	0.8325
	2	LPQ_{11}	RF	Samples converted to the greyscale	0.7847
	3	$LBP_{8,2}$	SVM	Border enhancement filter = *Scharr*	0.8277
	4	InceptionV3	SVM	# of *features* = 4096	0.8468
	5	VGG16	SVM	# of *features* = 4096	0.8421
	6	LeNet5	LeNet5	*Learning Rate* = 10^{-3}	0.7224
TW	1	LPQ_{13}	SVM	–	0.9712
	2	LPQ_{13}	RF	–	0.8920
	3	$RLBP_{8,2}$	SVM	Border enhancement filter = *Sobel*	0.9471
	4	VGG16	SVM	# of *features* = 2048	0.9327
	5	AlexNet	AlexNet	*Learning Rate* = 10^{-4}	0.8721
	6	MaxNet	MaxNet	*Learning Rate* = 10^{-3}	0.8199
D	1	LPQ_{13}	SVM	Pseudo-coloring = HSV	0.8967
	2	LPQ_{13}	RF	Pseudo-coloring = HSV	0.8325
	3	LPQ_{13}	SVM	Border enhancement filter = *Laplacian*	0.8695
	4	VGG16	SVM	# of *features* = 4096	0.9043
	5	AlexNet	AlexNet	*Learning Rate* = 10^{-5}	0.7475
	6	MaxNet	MaxNet	*Learning Rate* = 10^{-4}	0.7216

three different types of feature representation are ranked per disease group (from first to last), and an average rank is calculated. Then a general average ranking is computed for each one of the three feature representation scenarios.

The average rankings obtained with the previously described statistical test for the handcrafted features are presented in Table 13. We may observe that for the AIA disease, the best-ranked feature with an average ranking of 1.50 was LBP, while for the TW and D diseases, the best-ranked feature was LPQ, with an average ranking of 1.00. Moreover, considering all disease groups, the best ranked handcrafted feature is LPQ, with an average of 1.44.

Table 14 shows the average rankings for the non-handcrafted features obtained with pre-configured CNNs. We may observe that in all disease groups, AlexNet obtained the best classification results, reaching an average ranking of 1.50 for the AIA disease and 1.00 for TW and D diseases. Therefore, AlexNet is also the best-ranked feature in general, i.e., considering the three disease groups.

Table 15 presents the average ranking for the non-handcrafted features obtained with transfer learning. We

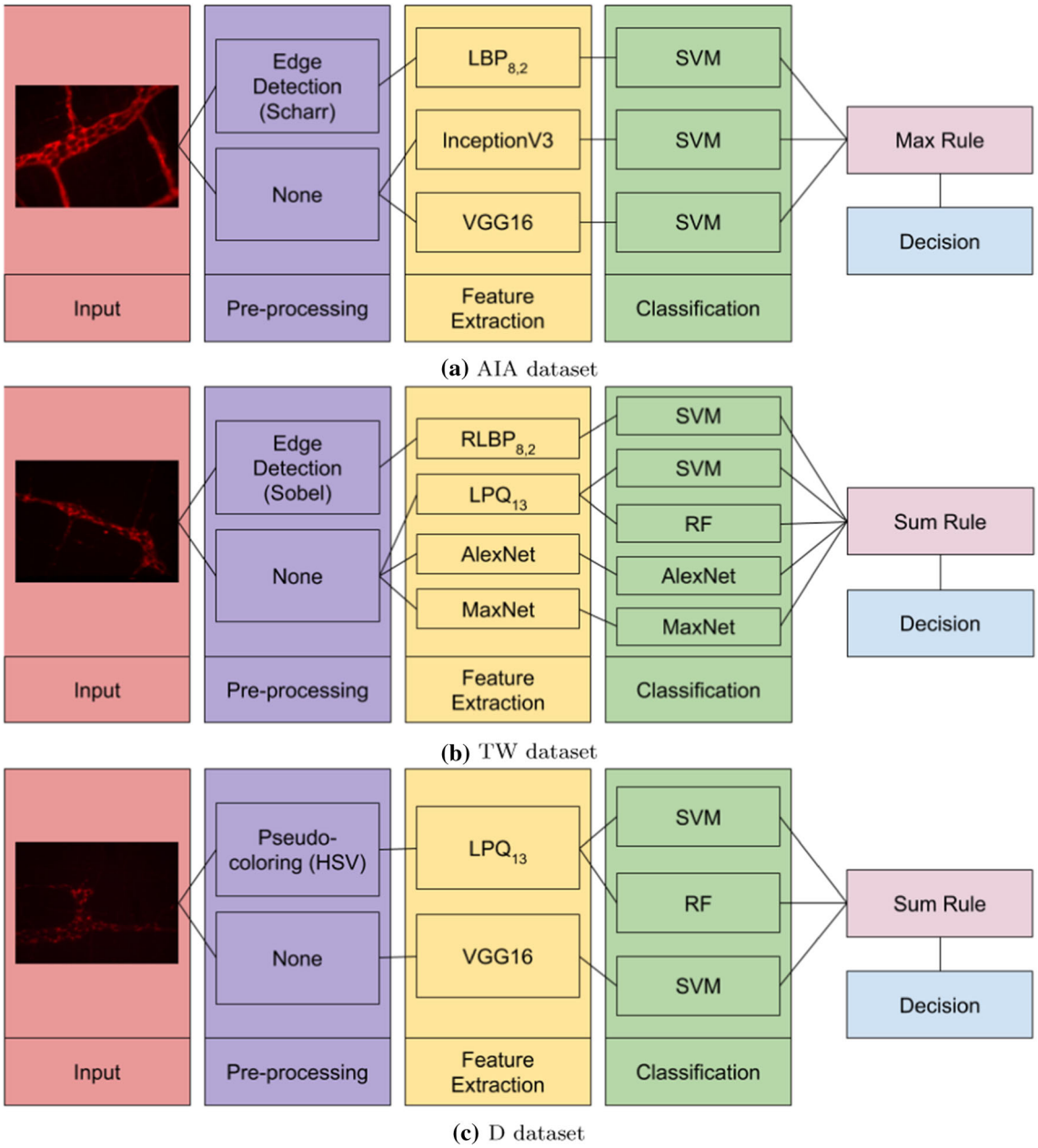

Fig. 11 Graphical representation of the configurations architected, so the best classification rates of this work could be achieved, by combining classifiers

may note that VGG16 achieved the best overall average ranking considering all disease groups (1.13), with an average ranking of 1.40 for AIA and 1.00 for both TW and D disease groups.

5.2 Which feature representation provided the best results?

To answer the second question with statistical significance, we applied the Wilcoxon Statistical Test. We hypothesize that the top-10 f-measures obtained with the handcrafted features are higher than the top-10 f-measures obtained with the non-handcrafted features (considering both pre-configured CNNs and transfer learning scenarios). The test was computed three times, one for each disease group.

The *z-scores* and *p-values* obtained with the Wilcoxon Statistical Test are reported in Table 16. Considering a threshold of 0.05, we can statistically affirm that for TW and D disease groups, the handcrafted features obtained better classification results than the non-handcrafted features, since their *p-values* are below the threshold with 0.0025 and 0.0035 values, respectively. However, considering the same threshold, we cannot statistically confirm that the handcrafted features are better than the non-handcrafted for the AIA disease, since its *p-value* resulted in 0.1931.

5.3 Which disease group is easier/harder to predict?

To answer the third question, we have also used the Wilcoxon Statistical Test. We have applied the test hypothesizing that the f-measures classification results obtained in all experiments for a certain disease group are lower than the f-measures obtained in another disease group. We have computed the test three times since we have three different disease groups. The *z-scores* and *p-values* obtained with the Wilcoxon Statistical Test for this scenario are reported in Table 17.

Considering a threshold of 0.05, we can statistically affirm that the classification results obtained in AIA disease are lower than the results obtained in TW disease, since the test achieved a *p-value* of $6.5 \times e^{-12}$ (below the threshold). We can also affirm that the results obtained in D disease are lower than the results obtained in TW, since the test obtained a *p-value* of $7.0 \times e^{-11}$. Thus, we can conclude that TW disease is the "easiest" one to predict, as it achieved classification results higher than the two other disease groups, i.e., AIA and D.

Furthermore, we can also observe in Table 17 that the classification results obtained in AIA disease are lower then the ones obtained in D disease group, since the test achieved a *p-value* of $7.0 \times e^{-4}$ (below the threshold). Thus, we can conclude that AIA disease is the "hardest" one to predict, as it achieved classification results lower than the two other disease groups (D and TW).

5.4 Have the fusion strategies contributed to improve the results?

To answer this question, our hypothesis is that the top-5 classification results with the combinations of the classifiers' outputs are higher than the top-5 classification results without the combination of outputs in each disease group.

Table 13 Average ranking of the classification results for the handcrafted features

	Disease			Overall Avg. Ranking
	AIA	TW	D	
LBP	1.50	2.50	2.33	2.11
RLBP	2.00	2.17	2.67	2.28
LPQ	2.33	1.00	1.00	1.44

Table 14 Average ranking of the classification results for the non-handcrafted features obtained with pre-configured CNNs

	Disease			Overall Avg. Ranking
	AIA	TW	D	
MaxNet	2.50	2.00	2.00	2.17
LeNet5	2.00	3.00	3.00	2.67
AlexNet	1.50	1.00	1.00	1.17

Table 15 Average ranking of the classification results for the non-handcrafted features obtained with transfer learning

	Disease			Overall Avg. Ranking
	AIA	TW	D	
VGG16	1.40	1.00	1.00	1.13
InceptionV3	1.60	2.00	2.00	1.87
InceptionResNetV2	3.00	3.00	3.00	3.00

Table 16 Wilcoxon statistical tests for the top-10 classification results with handcrafted features versus non-handcrafted

Disease	*z-score*	*p-value*
AIA	19	0.1931
TW	0	0.0025
D	1	0.0035

The *z-scores* and *p-values* obtained with the Wilcoxon Statistical Test for this scenario are reported in Table 18.

Considering a threshold of 0.05, we can statistically affirm that the fusion strategies did contribute to improving the overall classification results, since the *p-values* of the Wilcoxon test achieved 0.0216 for all disease groups.

6 Concluding remarks and future works

Nowadays, as the pre-clinical research advances, it is created the tendency of building methods capable of supporting such activities, aiming to decrease the manual work demanded and automatically analyze the conceived data. Thus, we proposed a method to automatically identifying chronic degenerative diseases using EGC image samples. Two types of features were considered (handcrafted and non-handcrafted). Both methodologies had their efficiency measured. Furthermore, a hybrid methodology was evaluated. Classifiers formed from such methodologies were combined to ascertain if it is possible to take advantage of a potential complementarity between them.

By experiments that were performed using handcrafted features, it was possible to observe a tendency of better results when the LPQ texture descriptor and the SVM classification algorithm were used. This descriptor's efficiency can be justified by the fact that it was developed aiming to be efficient for blurred images, like the ones existing in this work's dataset. At the end of these experiments' execution, we may observe that the best F-measure values founded reached 0.8325, 0.9712 and 0.8967 for AIA, TW, and D datasets, respectively. It is worth mentioning that for the AIA and D image samples, a pre-processing was applied, in which these were converted to grayscale (AIA) and pseudo-colored to the HSV colormap (D).

In the second approach used, non-handcrafted features were extracted using Feature Learning methods. Experiments were accomplished using CNN architectures and presented regular classification rates, but inferior to those found until that point. Better classification rates were generated by using a Transfer Learning strategy. The tests that led such results, extracted features using CNN architectures pre-trained with the ImageNet dataset, performing the classification later with the SVM classification algorithm. The best F-measure values found at the end of these tests were 0.8468, 0.9327, and 0.9043 for AIA, TW, and D datasets, respectively. Therefore, a slight improvement may be observed compared to the best ones, found by the classification of handcrafted features, for the AIA and D datasets.

Lastly, classifiers from both approaches were combined. Different configurations/architectures were created, and the target classifiers were combined by the sum, max, and product rules. By the use of this approach, this work's best results were obtained. The final F-Measure values were equal to 0.8930, 0.9845, and 0.9513 to the AIA, TW, and D datasets, respectively. These results were obtained by combining respectively three, five, and three classifiers. In all of them, classifiers created using both handcrafted and non-handcrafted features.

At the end of this work, it was possible to conclude that the classification of EGC images, aiming to identify a target disease's presence in the data samples, presented great results by classifying features extracted both on handcrafted and non-handcrafted modes. However, combining both strategies achieved the best performances for all disease groups. This corroborates the effectiveness of the proposed method.

In future works, we intend to expand the dataset aiming to include other types of cells, such as the Enteric Neuron. It could be useful to create an alternative understanding of the studied diseases and evaluate if it can improve the results described in this work.

We also intend to use the data samples as input in a segmentation methodology, using the isolated EGC to create a morphometric and quantitative analysis, aiming to perform the classification more precisely. We plan to adapt our method to analyze samples of individuals under different treatments so that it can i) automatically identify the proximity level to both classes approached in this work (control/healthy and unhealthy); ii) rank the performed treatments, displaying the most successful ones. Finally, we plan to: i) test new data augmentation methods; ii) crate a dedicated CNN architecture; iii) evaluate other texture descriptors; and iv) test different feature selection methods.

Table 17 Wilcoxon statistical tests comparing the classification results obtained in the different diseases groups

	z-score	*p-value*
AIA versus TW	143	$6.5 \times e^{-12}$
D versus TW	210	$7.0 \times e^{-11}$
AIA versus D	820	$7.0 \times e^{-4}$

Table 18 Wilcoxon statistical tests comparing the classification results before and after combining the classifiers outputs

	z-score	*p-value*
AIA	0.0	0.0216
TW	0.0	0.0216
D	0.0	0.0216

Declarations

Conflicts of interest The authors declare that they have no conflict of interest.

References

1. Bengio Y, Courville A, Vincent P (2013) Representation learning: a review and new perspectives. In: IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 1798–1828
2. Bossolani GDP et al (2019) Rheumatoid arthritis induces enteric neurodegeneration and jejunal inflammation, and quercetin promotes neuroprotective and anti-inflammatory actions. Life Sci
3. Chang C, Lin C (2013) Libsvm: a library for support vector machines. National Taiwan University, Taipei
4. Charte F, Rivera A, del Jesus M, Herrera F (2015) MLSMOTE: approaching imbalanced multilabel learning through synthetic instance generation. Knowl Based Syst 89:385–397
5. Chen, J, et al. (2013) RLBP: Robust local binary pattern, Center for machine vision research. University of Oulu, Oulu
6. Chollet F, et al. (2015) Keras. https://keras.io
7. Costa YMG (2013) Reconhecimento de gêneros musicais utilizando espectrogramas com combinação de classificadores. Ph.D. thesis, Federal University of Paraná, Curitiba, Brasil
8. Costa YMG, Oliveira LES, Koerich AL, Gouyon F, Martins JG (2012) Music genre classification using lbp textural features. Sig Process 92(11):2723–2737
9. Costa YMG, Oliveira LES, Silla CN Jr (2017) An evaluation of convolutional neural networks for music classification using spectrograms. Appl Soft Comput 52:28–38
10. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L (2009) ImageNet: a large-scale hierarchical image database. In: CVPR09, p. 8
11. Duda RO, Hart PE, Stork DG (2001) Pattern classification, 2nd edn. Wiley, New York
12. Felipe GZ, Aguiat RL, Costa YMG, Silla CN, Brahnam S, Nanni L, McMurtrey S (2019) Identification of infants' cry motivation using spectrograms. In: Proceedings of the international conference on systems, signals and image processing, pp. 181–186
13. Felipe GZ, Costa YMG, Helal LG (2017) Acoustic scene classification using spectrograms. In: Proceedings of the international conference of the Chilean computer science society, p. 7
14. Freitas GK, Costa YMG, Aguiar RL (2016) Using spectrogram to detect north atlantic right whale calls from audio recordings. In: Proceedings of the international conference of the Chilean computer science society, pp. 1–6
15. Frez FCV et al (2017) Restoration of density of interstitial cells of cajal in the jejunum of diabetic rats after quercetin supplementation. Rev Esp Enferm Dig 109:190–195
16. Furness JB (2006) The enteric nervous system. Blackwell Publishing, New Jersey, p 290
17. Giorgio RD et al (2012) Enteric glia and neuroprotection: basic and clinical aspects. Am J Physiol Gastrointest Liver Physiol 303:G887–G893
18. Gulbransen B, Sharkey K (2012) Novel functional roles for enteric glia in the gastrointestinal tract. Gastroenterol Hepatol. Vol, Nature reviews, p 9
19. Ho TK (1995) Random decision forests. In: Proceedings of the international conference on document analysis and recognition, pp. 278–282
20. Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. In: Proceedings of the international conference for learning representations, p. 15
21. Kittler J, Hatef M, Duin RP, Matas J (1998) On combining classifiers. IEEE Trans Pattern Anal Mach Intell 20(3):226–239
22. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Pereira F, Burges CJS, Bottou L, Weinberger KQ (Eds) Advances in Neural information processing systems, pp. 1097–1105. Curran Associates, Inc
23. Lecun Y (1989) Generalization and network design strategies. In: Pfeifer R, Schreter Z, Fogelman F, Steels L (eds) Connectionism in perspective. Elsevier, Amsterdam
24. Lecun Y et al (1989) Backpropagation applied to handwritten zip code recognition. Neural Comput 1:541–551
25. Liu T, Fang S, Zhao Y, Zhang J (2015) Implementation of training convolutional neural networks. Computing Research Repository (CoRR) **abs/1506.01195**
26. Mäenpää TI (2013) The local binary pattern approach to texture analysis: extensions and applications. Ph.D. thesis, University of Oulu, Oulu
27. Matsushita GHG, Sugi AH, Costa YMG, Gomez-A A, Da Cunha C, Oliveira LES (2019) Phasic dopamine release identification using convolutional neural network. Comput Biol Med 114:103466
28. Nanni L, Costa YMG, Lucio DR, Silla CN, Brahnam S (2016) Combining visual and acoustic features for bird species classification. In: Proceedings of the IEEE international conference on tools with artificial intelligence, pp. 396–401
29. Nanni L, Ghidoni S, Brahnam S (2017) Handcrafted vs non-handcrafted features for computer vision classitcation. Pattern Recogn 71:158–172. https://doi.org/10.1016/j.patcog.2017.05.025
30. Natekin A, Knoll A (2013) Gradient boosting machines, a tutorial. Front Neurorobotics 7:21
31. Ojala T, Pietikäinen M, Mäenpää T (2002) Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans Pattern Anal Mach Intell 24(7):971–987
32. Ojansivu V, Heikkilä J (2008) Blur insensitive texture classification using local phase quantization. Image and Signal Processing pp. 236–243
33. Pan SJ, Yang Q (2010) A survey on transfer learning. IEEE Trans Knowl Data Eng 22(10):1345–1359
34. Panizzon CPNB et al (2016) Desired and side effects of the supplementation with l-glutamine and l-glutathione in enteric glia of diabetic rats. Acta Histochem 118:625–631
35. Panizzon CPNB et al (2019) Ethyl acetate fraction from trichilia catigua confers partial neuroprotection in components of the enteric innervation of the jejunum in diabetic rats. Cell Physiol Biochem 53:76–86
36. Paulino MAD, Britto Junior AS, Svaigen, AR, Aylon LBR, Oliveira LES, Costa YMG (2018) A brazilian speech database. In: Proceedings of the IEEE international conference on tools with artificial intelligence, pp. 234–241
37. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E (2011) Scikit-learn: machine learning in Python. J Mach Learn Res 12:2825–2830
38. Pereira RM, Bertolini D, Teixeira LO, Silla CN Jr, Costa YMG (2020) Covid-19 identification in chest x-ray images on flat and hierarchical classification scenarios. Comput Methods Programs Biomed 194:105532
39. Pereira RV et al (2011) L-glutamine supplementation prevents myenteric neuron loss and has gliatrophic effects in the ileum of diabetic rats. Dig Dis Sci 56:3507–3516
40. Roecker MN, Costa YMG, Almeida JLR, Matsushita G (2018) Automatic vehicle type classification with convolutional neural

networks. In: Proceedings of the international conference on systems, signals and image processing, pp. 1–5
41. Ruhl A (2005) Glial cells in the gut. Neurogastroenterol Motil 17:777–790
42. Russell SJ, Norvig P (2010) Artificial intelligence: a modern approach, 3rd edn. Pearson Education, London
43. Schapire RE (1999) A brief introduction to boosting. In: Proceedings of the 16th international joint conference on artificial intelligence - Volume 2, IJCAI'99, p. 1401–1406. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA
44. Sharkey KA (2015) Emerging roles for enteric glia in gastrointestinal disorders. J Clin Investig 125
45. Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. Computing Research Repository (CoRR) **abs/1409.1556**
46. Souza ID et al (2011) Analysis of myosin-v immunoreactive myenteric neurons from arthritic rats. Arq Gastroenterol 48:205–210
47. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 15:1929–1958
48. Szegedy C, Ioffe S, Vanhoucke V (2016) Inception-v4, inception-resnet and the impact of residual connections on learning. Comput Res Repos (CoRR) **abs/1602.07261**
49. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2015) Rethinking the inception architecture for computer vision. Comput Res Repos (CoRR) **abs/1512.00567**
50. Vargas ACG, Paes AVCN (2016) Um estudo sobre redes neurais convolucionais e sua aplicação em detecção de pedestres. In: Proceedings of the conference on graphics, patterns and images, p. 4
51. Vicentini GE et al (2017) Does l-glutamine-supplemented diet extenuate no-mediated damage on myenteric plexus of walker 256 tumor-bearing rats? Food Res Int 101:24–34
52. Zhao Y, Jia W, Hu RX, Min H (2013) Completed robust local binary pattern for texture classification. Neurocomputing 106:68–76